Acta Geotechnica (2006) 1:237–252
DOI 10.1007/s11440-006-0019-3

RESEARCH PAPER

Modeling bed erosion in free surface flows by the particle finite element method

Eugenio Oñate · Miguel A. Celigueta · Sergio R. Idelsohn

Received: 1 September 2006 / Accepted: 10 October 2006 / Published online: 21 November 2006

Abstract We present a general formulation for modeling bed erosion in free surface flows using the particle finite element method (PFEM). The key feature of the PFEM is the use of an updated Lagrangian description to model the motion of nodes (particles) in domains containing fluid and solid subdomains. Nodes are viewed as material points (called particles) which can freely move and even separate from the fluid and solid subdomains representing, for instance, the effect of water drops or soil/rock particles. A mesh connects the nodes defining the discretized domain in the fluid and solid regions where the governing equations, expressed in an integral form, are solved as in the standard FEM. The necessary stabilization for dealing with the incompressibility of the fluid is introduced via the finite calculus (FIC) method. An incremental iterative scheme for the solution of the nonlinear transient coupled fluid-structure problem is described. The erosion mechanism is modeled by releasing the material adjacent to the bed surface according to the frictional work generated by the fluid shear stresses. The released bed material is subsequently transported by the fluid flow. Examples of application of the PFEM to solve a number of bed erosion problems involving large motions of the free surface and splashing of waves are presented.

E. Oñate (✉) · M. A. Celigueta · S. R. Idelsohn
International Center for Numerical Methods in Engineering (CIMNE), Universidad Politecnica de Cataluña, Barcelona 08034, Spain
e-mail: onate@cimne.upc.edu
URL: http://www.cimne.com/pfem

Keywords Bed erosion · Free surface flows · Particle finite element method

1 Introduction

Modeling bed erosion and sediment transport in open channel flows is an extremely important task in many areas of river and environmental engineering and related areas. For example, bed erosion can lead to instabilities of the river basin slopes. It can also undermine the foundation of bridge piles thereby favouring structural failure. Modeling of bed erosion is also relevant for predicting the evolution of surface material dragged in earth dams in overspill situations. Bed erosion is one of the main causes of environmental damage in floods.

Prediction of erosion of soil/rock particles in bed surfaces due to water streams is very difficult due to the complexity of accurately predicting the tangential stresses at the fluid-bed interface which are mainly responsible for the detachment of bed particles. Definition of the erosion onset conditions for different bed geomaterials is also an important and difficult task. Modeling of sediment transport phenomena is also very complex.

The erosion and transport of sediment particles in environmental flows can be analyzed by solving the Navier–Stokes equations for the water flow, either in the fully 3D version, or via a simpler 2D depth average model in combination with an assumed vertical velocity profile (typically of logarithmic type). The flow field variables computed at each time step, or at every iteration within a time step in a strongly coupled scheme, are used as input data for solving the sediment

transport equations and other relevant information such as the suspended load concentration, the bed load transport rate and the bed deformation. Numerical solutions for these type of problems have been reported using mainly finite difference and finite volume schemes in Eulerian and arbitrary Lagrangian-Eulerian (ALE) grids for solving both the fluid flow and the sediment transport equations [4, 7, 14, 26–34].

In addition to the intrinsic complexities of the multiphysics flow-erosion-transport problem, the numerical solution of the equations for the fluid–structure interaction (FSI) problem in free surface flows is faced with the treatment of the convective terms and the incompressibility constraint in the fluid equations, the modeling of the free surface accounting for wave splashing, the transfer of information between the fluid and solid domains via the contact interfaces and the tracking of solid elements within the fluid domain. Indeed most of these problems are extremely difficult to model using the Eulerian and ALE formulations.

An alternative approach which simplifies many of above difficulties is to use a Lagrangian description to formulate the governing equations of both the solid and the fluid domains. In the Lagrangian formulation the motion of the individual particles are followed and, consequently, nodes in a finite element mesh can be viewed as moving material points (hereforth called "particles"). Hence, the motion of the mesh discretizing the total domain (including both the fluid and solid regions) is followed during the transient solution.

In this paper we present a particular class of Lagrangian formulation developed by the authors to solve bed erosion problems in free surface flows. The method is an extension of the so-called particle finite element method (PFEM). The PFEM treats the mesh nodes in the fluid and solid domains as particles which can freely move and even separate from each domain representing, for instance, the effect of water drops or solid particles. A finite element mesh connects the nodes defining the discretized domain where the governing equations are solved in the standard FEM fashion. The particular application of the PFEM to model bed erosion problems here described is the natural evolution of recent work of the authors for the solution of FSI problems using Lagrangian finite element and meshless methods [2, 10, 11, 13, 21–23].

An obvious advantage of the Lagrangian formulation is that the convective terms disappear from the fluid equations. The difficulty is, however, transferred to the problem of adequately (and efficiently) moving the mesh nodes. Indeed in the PFEM approach remeshing is a frequent necessity along the time solution. We use an innovative mesh regeneration procedure blending elements of different shapes using an extended Delaunay tesselation with adequate C° continuous shape functions [10, 12].

The need to properly treat the incompressibility condition in the fluid still remains in the Lagrangian formulation. The use of standard finite element interpolations may lead to a volumetric locking defect unless some precautions are taken [3, 5, 35]. In our work volumetric locking is avoided via a FIC procedure [17].

The layout of the paper is the following. In the next section the basic ideas of the PFEM are outlined. Next the basic equation for an incompressible flow using a Lagrangian description and the FIC formulation are presented. Then a fractional step scheme for the transient solution via standard finite element procedures is described. Details of the treatment of the coupled FSI problem are given. The procedures for mesh generation, for identification of the free surface nodes, for treating frictional contact situations and for modeling bed erosion are described. Finally, the efficiency of the PFEM is shown in its application to a number of bed erosion problems involving surface waves.

2 The basis of the PFEM

Let us consider a continuum domain containing both fluid and solid subdomains. The fluid particles interact with the solid boundaries thereby inducing the deformation of the solid which in turn affects the flow motion and, therefore, the problem is fully coupled.

In the PFEM approach, both the fluid and the solid domains are modeled using an updated Lagrangian formulation. That is, all variables in the fluid and solid domains are assumed to be known in the current configuration at time t. The new set of variables in both domains are sought for in the next or updated configuration at time $t + \Delta t$ (Fig. 1). The finite element method (FEM) is used to solve the continuum equations in both domains. Hence a mesh discretizing these domains is generated in order to solve the governing equations for both the fluid and solid problems in the standard FEM fashion. We note again that the nodes discretizing the fluid and solid domains are viewed as material particles whose motion is tracked during the transient solution. This is useful to model the separation of fluid particles from the main fluid domain, or the separation of solid particles from the bed surface, and to follow their subsequent motion as individual particles with a known density, an initial acceleration and velocity and subject to gravity forces.

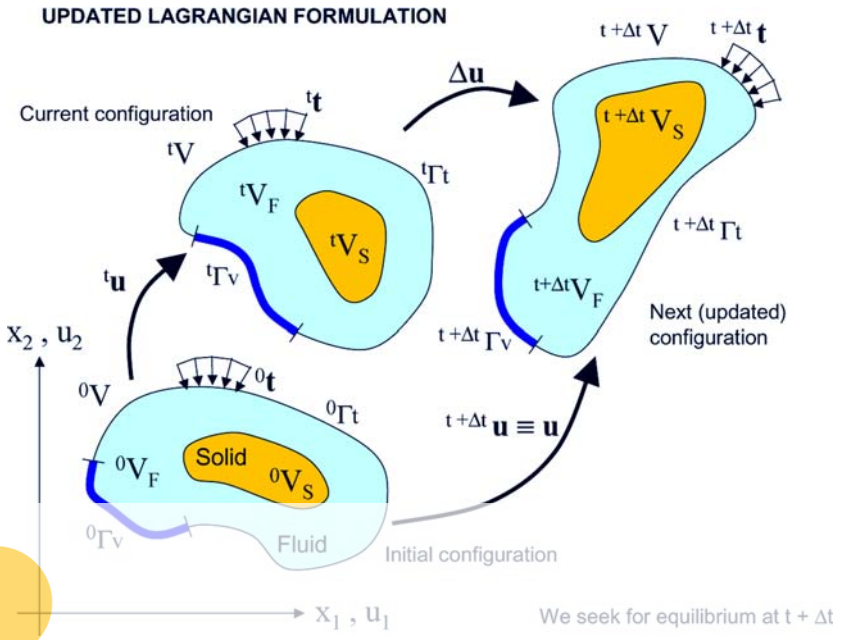

Fig. 1 Updated Lagrangian description for a continuum containing a fluid and a solid domain

It is important to recall that each particle is treated as a material point characterized by the density of the solid or fluid domain to which it belongs. The mass of a given domain is obtained by integrating the density at the different material points over the domain.

The quality of the numerical solution depends on the discretization chosen as in the standard FEM. Adaptive mesh refinement techniques can be used to improve the solution in zones where large gradients of the fluid or the structure variables occur.

2.1 Basic steps of the PFEM

For clarity purposes we will define the collection or cloud of nodes (C) pertaining to either the fluid and the solid subdomains, the volume (V) defining the analysis domain for the fluid and the solid, and the mesh (M) discretizing both domains.

A typical solution with the PFEM involves the following steps:

1. The starting point at each time step is the cloud of points in the fluid and solid subdomains. For instance ${}^{n}C$ denotes the cloud at time $t = t_n$ (Fig. 2).
2. Identify the boundaries for both the fluid and solid domains defining the analysis domain ${}^{n}V$ in the fluid and the solid. This is an essential step as some boundaries, such as the free surface in fluids or the bed surface, may be severely distorted during the solution process including separation and re-entering of nodes. The Alpha shape method [6] is used for the boundary definition (Sect. 7).
3. Discretize the fluid and solid subdomains with a finite element mesh ${}^{n}M$. In our work we use an innovative mesh generation scheme based on the extended Delaunay tesselation (Sect. 6) [10, 11, 13].
4. Solve the coupled Lagrangian equations of motion for the fluid and the solid domains. Compute the relevant state variables in both domains at the next (updated) configuration for $t + \Delta t$: velocities, pressure and viscous stresses in the fluid and displacements, stresses and strains in the solid. An overview of the coupled FSI algorithm is given in the next section.
5. Compute the frictional work (W_f) performed by the tangential stresses at the bed surface. Bed erosion initiates if W_f exceeds a critical value W_c. Bed surface points where $W_f > W_c$ are released from the bed domain and are subsequently transported by the fluid velocity.
6. Move the mesh nodes to a new position ${}^{n+1}C$ where $n + 1$ denotes the time $t_n + \Delta t$, in terms of the time increment size. This step is typically a consequence of the solution process of step 4. Recall that a node identifies a material point in either the fluid or solid subdomains.
7. Go back to step 1 and repeat the solution process for the next time step.

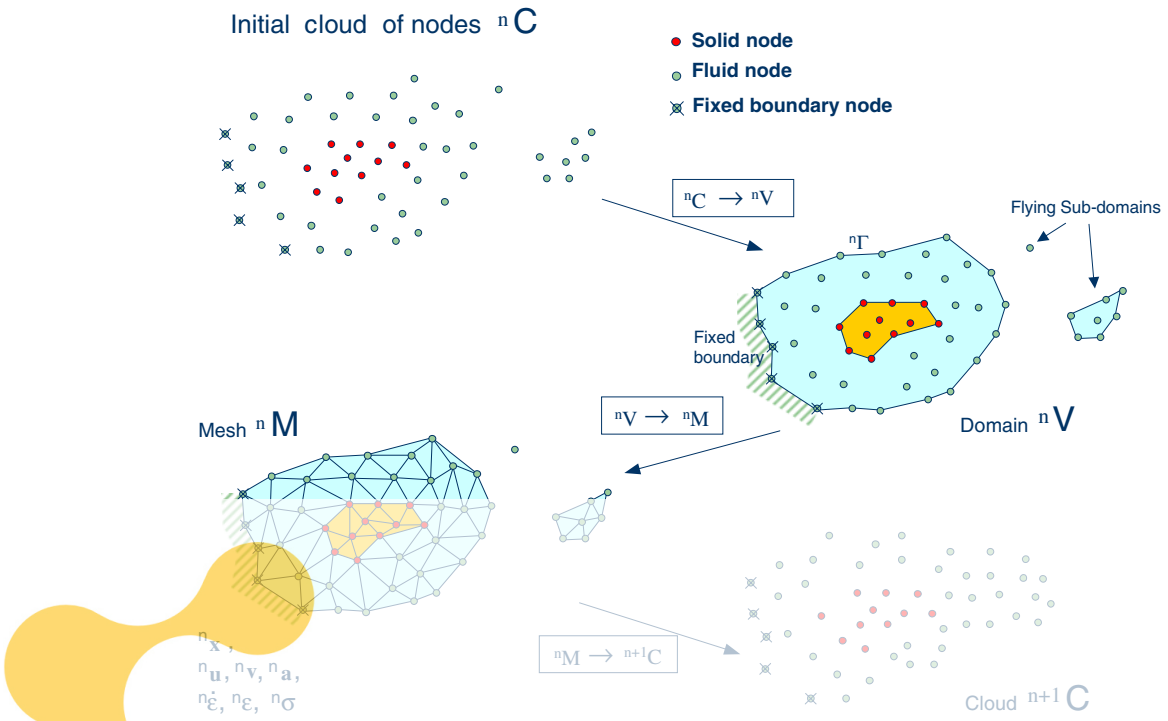

Fig. 2 Sequence of steps to update a "cloud" of nodes from time n ($t = t_n$) to time $n + 1$ ($t = t_n + \Delta t$)

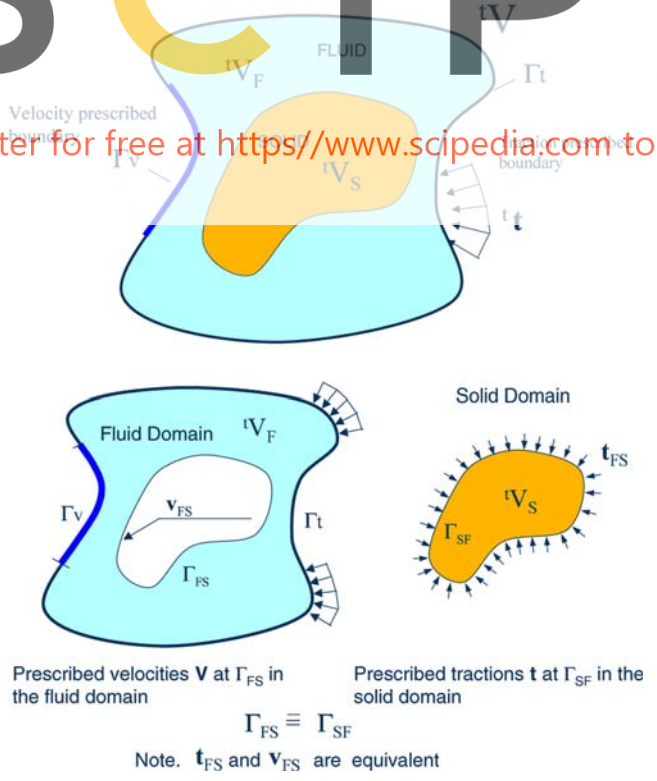

Fig. 3 Split of the analysis domain V into fluid and solid subdomains. Equivalence of surface tractions and kinematic variables at the common interface

2.2 Overview of the coupled FSI algoritm

Figure 3 shows a typical domain V with external boundaries Γ_V and Γ_t where the velocity and the surface tractions are prescribed, respectively. The domain V is formed by fluid (V_F) and solid (V_S) subdomains. Both subdomains interact at a common boundary Γ_{FS} where the surface tractions and the kinematic variables (displacements, velocities and acelerations) are the same for both subdomains. Note that both sets of variables (the surface tractions and the kinematic variables) are equivalent in the equilibrium configuration.

Note that the flow in an open channel is a particular case of above situations where the solid domain constitutes the bed region whose surface is eroded by the interaction with the fluid particles in motion.

Let us define ${}^tS$ and ${}^tF$ as the set of variables defining the kinematics and the stress–strain fields in the solid and fluid domains at time t, respectively, i.e.

$$ {}^tS := [{}^t\mathbf{x}_s, {}^t\mathbf{u}_s, {}^t\mathbf{v}_s, {}^t\mathbf{a}_s, {}^t\boldsymbol{\varepsilon}_s, {}^t\boldsymbol{\sigma}_s, \ldots]^T \tag{1} $$

$$ {}^tF := [{}^t\mathbf{x}_F, {}^t\mathbf{u}_F, {}^t\mathbf{v}_F, {}^t\mathbf{a}_F, {}^t\dot{\boldsymbol{\varepsilon}}_F, {}^t\boldsymbol{\sigma}_F, \ldots]^T \tag{2} $$

where $\mathbf{x}$ is the nodal coordinate vector, $\mathbf{u}$, $\mathbf{v}$ and $\mathbf{a}$ are the vector of displacements, velocities and accelerations, respectively, $\boldsymbol{\varepsilon}, \dot{\boldsymbol{\varepsilon}}, \boldsymbol{\sigma}$ are the strain vector, the

strain-rate (or rate of deformation) vectors and the Cauchy stress vector, respectively and F and S denote the variables in the fluid and solid domains, respectively. In the discretized problem, a bar over these variables will denote nodal values.

The coupled FSI problem of Fig. 3 is solved using the following conceptual scheme:

0. We assume that the variables in the solid and fluid domains at time t (^tS and ^tF) are known.
1. Solve for the variables at the solid domain at time $t + \Delta t$ ($^{t+\Delta t}S$) under prescribed surface tractions at the fluid–solid boundary Γ_{FS}.
2. Solve for the variables at the fluid domain at time $t + \Delta t$ ($^{t+\Delta t}F$) under prescribed surface tractions at the external boundary Γ_t and prescribed velocities at the external and internal boundaries Γ_V and Γ_{FS}, respectively.

Iterate between 1 and 2 until convergence.

The variables at the solid domain $^{t+\Delta t}S$ are found via the integration of the dynamic equations of motion in the solid region written as

$$\mathbf{M}_s\mathbf{a}_s + \mathbf{g}_s - \mathbf{f}_s = \mathbf{0} \tag{3}$$

where $\mathbf{M}_s$, $\mathbf{g}_s$ and $\mathbf{f}_s$ denote the mass matrix, the internal node force vector and the external nodal force vector in the solid domain. The time integration of Eq. 3 is performed using a standard Newmark method. An incremental iterative scheme is implemented within each time step to account for nonlinear geometrical and material effects [36].

The FEM solution of the variables in the (incompressible) fluid domain implies solving the momentum and incompressibility equations. In our work we use a stabilized FEM based on the FIC approach which allows to use a linear approximation for the velocity and pressure variables [15, 17]. Details of the FEM/FIC formulation used are given in the next section.

Figure 4 shows a typical example of a PFEM solution in 2D. The pictures correspond to the analysis of the problem of breakage of a water column [13, 23]. Figure 4a shows the initial grid of four node rectangles discretizing the fluid domain and the solid walls. Boundary nodes identified with the Alpha-shape method have been marked with a circle. Figure 4b and c show the mesh for the solution at two later times.

3 FIC/FEM formulation for a Lagrangian incompressible fluid

The standard infinitesimal equations for a viscous incompressible fluid can be written in a Lagrangian frame as [15, 35].

$$\textbf{Momentum} \quad r_{mi} = 0 \quad \text{in } V_F \tag{4}$$

$$\textbf{Mass balance} \quad r_d = 0 \quad \text{in } V_F \tag{5}$$

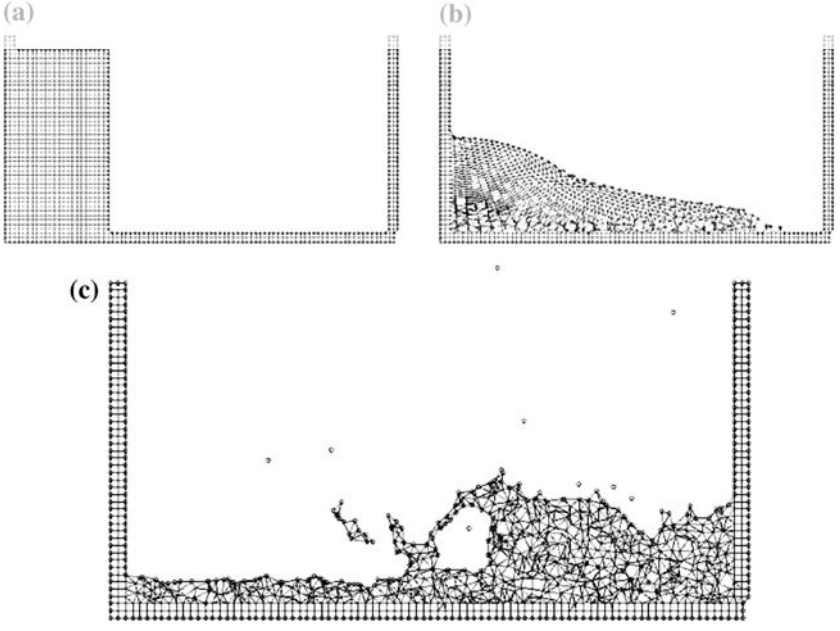

Fig. 4 Breakage of a water column. **a** Discretization of the fluid domain and the solid walls. Boundary nodes are marked with *circles*. **b** and **c** Mesh in the fluid and solid domains at two different times

where

$$r_{m_i} = \rho \frac{\partial v_i}{\partial t} - \frac{\partial \sigma_{ij}}{\partial x_j} - b_i, \quad \sigma_{ji} = \sigma_{ij} \tag{6}$$

$$r_d = \frac{\partial v_i}{\partial x_i} \quad i,j = 1, n_d \tag{7}$$

In the above, variable n_d is the number of space dimensions, v_i is the velocity along the ith global axis ($v_i = \partial u_i/\partial t$, where u_i is the ith displacement), ρ is the (constant) density of the fluid, b_i are the body forces, σ_{ij} are the total stresses given by $\sigma_{ij} = s_{ij} - \delta_{ij} p$, p is the absolute pressure (defined positive in compression) and s_{ij} are the viscous deviatoric stresses related to the viscosity μ by the standard expression

$$s_{ij} = 2\mu \left(\dot{\varepsilon}_{ij} - \delta_{ij} \frac{1}{3} \frac{\partial v_k}{\partial x_k} \right) \tag{8}$$

where δ_{ij} is the Kronecker delta and the strain rates $\dot{\varepsilon}_{ij}$ are

$$\dot{\varepsilon}_{ij} = \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) \tag{9}$$

In the above all variables are defined at the current time t (current configuration). The standard summation convention for repeated indexes is assumed unless otherwise specified.

In our work we will solve a modified set of governing equations derived using a FIC formulation. The FIC governing equations are [15, 16, 18]

Momentum $$r_{m_i} - \underline{\frac{1}{2} h_j \frac{\partial r_{m_i}}{\partial x_j}} = 0 \quad \text{in } V_F \tag{10}$$

Mass balance $$r_d - \underline{\frac{1}{2} h_j \frac{\partial r_d}{\partial x_j}} = 0 \quad \text{in } V_F \tag{11}$$

The problem definition is completed with the following boundary conditions:

$$n_j \sigma_{ij} - t_i + \underline{\frac{1}{2} h_j n_j r_{m_i}} = 0 \quad \text{on } \Gamma_t \tag{12}$$

$$v_j - v_j^p = 0 \quad \text{on } \Gamma_v \tag{13}$$

and the initial condition is $v_j = v_j^0$ for $t = t_0$.

In Eqs. 12 and 13, t_i and v_j^p are surface tractions and prescribed velocities on the boundaries Γ_t and Γ_v, respectively, n_j are the components of the unit normal vector to the boundary.

The h_i's in above equations are characteristic lengths of the domain where balance of momentum and mass is enforced. In Eq. 12 these lengths define the domain where equilibrium of boundary tractions is established. In our work we have taken h_i to be constant at each element and equal to a typical element dimension h^e computed as $h^e = [V^e]^m$ where V^e is the element volume and $m = 1/2$ for 2D problems and $m = 1/3$ for 3D problems. Details of the derivation of Eqs. 10–13 can be found in Oñate [15, 16, 18].

Equations 10–13 are the starting points for deriving stabilized finite element methods to solve the incompressible Navier–Stokes equations in a Lagrangian frame of reference using equal order interpolation for the velocity and pressure variables [2, 9– 11, 13, 21]. Application of the FIC formulation to finite element and meshless analysis of fluid flow problems can be found in [8, 16–19, 20, 22].

3.1 Transformation of the mass balance equation. Integral governing equations

The underlined term in Eq. 11 can be expressed in terms of the momentum equations. The new expression for the mass balance equation is [16, 23]

$$r_d - \sum_{i=1}^{n_d} \tau_i \frac{\partial r_{m_i}}{\partial x_i} = 0 \quad \text{with} \quad \tau_i = \frac{3h_i^2}{8\mu} \tag{14}$$

At this stage it is no longer necessary to retain the stabilization terms in the momentum equations. These terms are critical in Eulerian formulations to stabilize the numerical solution for high values of the convective terms. In the Lagrangian formulation the convective terms dissappear from the momentum equations and the FIC terms in these equations are just useful to derive the form of the mass balance equation given by Eq. 14 and can be disregarded there onwards. Consistently, the stabilization terms are also neglected in the Neumann boundary conditions (Eq.12).

The weighted residual expression of the final form of the momentum and mass balance equations can be written as

$$\int_{V_F} \delta v_i r_{m_i} \mathrm{d}V + \int_{\Gamma_t} \delta v_i (n_j \sigma_{ij} - t_i) \mathrm{d}\Gamma = 0 \tag{15}$$

$$\int_{V_F} q \left[r_d - \sum_{i=1}^{n_d} \tau_i \frac{\partial r_{m_i}}{\partial x_i} \right] \mathrm{d}V = 0 \tag{16}$$

where δv_i and q are arbitrary weighting functions equivalent to virtual velocity and virtual pressure fields.

The computation of the residual terms in Eq. 16 is simplified if we introduce the pressure gradient projections π_i, defined as

$$\pi_i = r_{m_i} - \frac{\partial p}{\partial x_i} \tag{17}$$

We express now r_{m_i} in Eq. 16 in terms of the π_i which then become additional variables. The system of integral equations is therefore augmented in the necessary number of equations by imposing that the residual r_{m_i} vanishes within the analysis domain (in an average sense). We proceed next to integrate by parts the r_{m_i} term in Eq. 16 and the deviatoric stresses and the pressure terms within r_{m_i} in Eq.15. The final system of governing equation is

$$\int_{V_F} \left[\delta v_i \rho \frac{\partial v_i}{\partial t} + \delta \dot{\varepsilon}_{ij}(s_{ij} - \delta_{ij} p) \right] \mathrm{d}V - \int_{V_F} \delta v_i b_i \, \mathrm{d}V - \int_{\Gamma_t} \delta v_i t_i \, \mathrm{d}\Gamma = 0 \tag{18}$$

$$\int_{V_F} q \frac{\partial v_i}{\partial x_i} \mathrm{d}V + \int_{V_F} \sum_{i=1}^{n_d} \tau_i \frac{\partial q}{\partial x_i} \left(\frac{\partial p}{\partial x_i} + \pi_i \right) \mathrm{d}V = 0 \tag{19}$$

$$\int_{V_F} \delta \pi_i \tau_i \left(\frac{\partial p}{\partial x_i} + \pi_i \right) \mathrm{d}V = 0 \quad \text{no sum in } i \tag{20}$$

with $i,j,k = 1,n_d$. In Eq. 18 $\delta\dot{\varepsilon}_{ij}$ are virtual strain rates. In Eq. 20 $\delta\pi_i$ are appropriate weighting functions and the τ_i weights are introduced for symmetry reasons. Note that the boundary term resulting from the integration by parts of r_{m_i} in Eq. 16 has been neglected as the influence of this term in the numerical solution has been found to be negligible.

3.2 Finite element discretization

We choose equal order C° continuous interpolations of the velocities, the pressure and the pressure gradient projections π_i over each element with n nodes. The interpolations are written as

$$v_i = \sum_{j=1}^{n} N_j \bar{v}_i^j, \quad p = \sum_{j=1}^{n} N_j \bar{p}^j, \quad \pi_i = \sum_{j=1}^{n} N_j \bar{\pi}_i^j \tag{21}$$

where $(\cdot)^{\bar{j}}$ denotes nodal variables and N_j are the shape functions [35].

Substituting the approximations (21) into Eqs. 19–20 and choosing a Galerkin form with $\delta v_i = q = \delta\pi_i = N_i$ leads to the following system of discretized equations:

$$\mathbf{M}\dot{\bar{\mathbf{v}}} + \mathbf{K}\bar{\mathbf{v}} - \mathbf{G}\bar{\mathbf{p}} - \mathbf{f} = \mathbf{0} \tag{22a}$$

$$\mathbf{G}^{\mathrm{T}}\bar{\mathbf{v}} + \mathbf{L}\bar{\mathbf{p}} + \mathbf{Q}\bar{\boldsymbol{\pi}} = \mathbf{0} \tag{22b}$$

$$\mathbf{Q}^{\mathrm{T}}\bar{\mathbf{p}} + \hat{\mathbf{M}}\bar{\boldsymbol{\pi}} = \mathbf{0} \tag{22c}$$

The matrices and vectors in Eqs. 22a, 22b, 22c are assembled from the element contributions given by (for 2D problems)

$$\mathbf{M}_{ij} = \int_{V_F^e} \rho \mathbf{N}_i \mathbf{N}_j \mathrm{d}V, \quad \mathbf{K}_{ij} = \int_{V_F^e} \mathbf{B}_i^{\mathrm{T}} \mathbf{D} \mathbf{B}_j \, \mathrm{d}V$$

$$\mathbf{D} = \mu \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{B}_i = \begin{bmatrix} \partial N_i/\partial x_1 & 0 \\ 0 & \partial N_i/\partial x_2 \\ \partial N_i/\partial x_2 & \partial N_i/\partial x_1 \end{bmatrix}$$

$$L_{ij} = \int_{V_F^e} \tau_k \frac{\partial N_i}{\partial x_k} \frac{\partial N_j}{\partial x_k} \mathrm{d}V, \quad \mathbf{Q} = [\mathbf{Q}^1, \mathbf{Q}^2],$$

$$Q_{ij}^k = \int_{V_F^e} \tau_k \frac{\partial N_i}{\partial x_k} N_j \mathrm{d}V$$

$$\hat{\mathbf{M}} = \begin{bmatrix} \hat{\mathbf{M}}^1 & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{M}}^2 \end{bmatrix}, \quad \hat{\mathbf{M}}_{ij}^k = \int_{V_F^e} \tau_k N_i N_j \mathrm{d}V,$$

$$\mathbf{G}_{ij} = \int_{V_F^e} \mathbf{B}_i^{\mathrm{T}} \mathbf{m} N_j \, \mathrm{d}V$$

$$\mathbf{f}_i = \int_{V_F^e} N_i \mathbf{b} \, \mathrm{d}V + \int_{\Gamma_t^e} N_i \mathbf{t} \, \mathrm{d}\Gamma, \quad \mathbf{b} = [b_1, b_2], \quad \mathbf{t} = [t_1, t_2] \tag{23}$$

with $i,j = 1,n$ and $k,l = 1,2$.

In above $\mathbf{B}$ is the strain rate matrix [35], V_F^e and Γ_t^e are the volume and the Neumann boundary of the element and $\mathbf{m} = [1,1,0]^{\mathrm{T}}$ for 2D problems.

3.3 Fractional step algorithm for the fluid variables

The starting point of the iterative algorithm are the variables at time n in the fluid domain (${}^nF$). The sought variables are the variables at time $n + 1$ (${}^{n+1}F$). For the sake of clarity we will skip the upper left index $n + 1$ for all variables, i.e.

$${}^{n+1}\bar{\mathbf{x}} \equiv \bar{\mathbf{x}}; \quad {}^{n+1}\bar{\mathbf{p}} \equiv \bar{\mathbf{p}}; \quad {}^{n+1}\bar{\boldsymbol{\pi}} \equiv \bar{\boldsymbol{\pi}}; \quad {}^{n+1}\bar{\mathbf{x}} \equiv \bar{\mathbf{x}}; \ldots. \tag{24}$$

A simple iterative algorithm is obtained by splitting the pressure from the momentum equations as follows:

$$\bar{\mathbf{v}}^{*} = {}^{n}\bar{\mathbf{v}} - \Delta t \mathbf{M}^{-1}[\mathbf{K}\bar{\mathbf{v}}^{j} - \mathbf{G}^{n}\mathbf{p} - \mathbf{f}] \tag{25}$$

$$\bar{\mathbf{v}}^{j+1} = \bar{\mathbf{v}}^{*} + \Delta t \mathbf{M}^{-1}\mathbf{G}\delta\bar{\mathbf{p}} \tag{26}$$

where $\delta\bar{\mathbf{p}}$ denotes a pressure increment. In above equations and in the following the left upper index n refers to values in the current configuration ${}^{n}V_{\mathrm{F}}$ whereas the right index j denotes the iteration number within each time step. The value of $\bar{\mathbf{v}}^{j+1}$ from Eqs. 26 is substituted now into Eq. 22b to give

$$\mathbf{G}^{\mathrm{T}}\bar{\mathbf{v}}^{*} + \Delta t \mathbf{S}\delta\bar{\mathbf{p}} + \mathbf{L}\bar{\mathbf{p}}^{j+1} + \mathbf{Q}\bar{\boldsymbol{\pi}}^{j} = \mathbf{0} \tag{27a}$$

where

$$\mathbf{S} = \mathbf{G}^{\mathrm{T}}\mathbf{M}^{-1}\mathbf{G} \tag{27b}$$

Typically matrix $\mathbf{S}$ is computed using a diagonal matrix $\mathbf{M} = \mathbf{M}_d$, where the subscript d denotes a diagonal matrix. Diagonalization can be performed by a lumping technique.

An alternative is to approximate matrix $\mathbf{S}$ by a Laplacian matrix. This reduces considerably the bandwidth of $\mathbf{S}$. The disadvantage is that the pressure increment must be prescribed on the free surface and this reduces the accuracy in the satisfaction of the incompressibility condition in these regions.

A semi-implicit algorithm can be derived as follows. For each iteration:

Step 1 Compute $\bar{\mathbf{v}}^{*}$ from Eq. 25 with $\mathbf{M} = \mathbf{M}_d$. For the first iteration $(\bar{\mathbf{v}}^{1}, \bar{\mathbf{p}}^{1}, \bar{\boldsymbol{\pi}}^{1}) = ({}^{n}\bar{\mathbf{v}}, {}^{n}\bar{\mathbf{p}}, {}^{n}\bar{\boldsymbol{\pi}})$.

Step 2 Compute $\delta\bar{\mathbf{p}}$ and $\bar{\mathbf{p}}^{j+1}$ from Eq. 27a as

$$\delta\bar{\mathbf{p}} = -(\mathbf{L} + \Delta t \mathbf{S})^{-1}[\mathbf{G}^{\mathrm{T}}\bar{\mathbf{v}}^{*} + \mathbf{Q}\bar{\boldsymbol{\pi}}^{j} + \mathbf{L}\bar{\mathbf{p}}^{j}] \tag{28a}$$

The pressure $\bar{\mathbf{p}}^{j+1}$ is computed as follows

$$\bar{\mathbf{p}}^{j+1} = \bar{\mathbf{p}}^{j} + \delta\bar{\mathbf{p}}^{j} \tag{28b}$$

Step 3 Compute $\bar{\mathbf{v}}^{j+1}$ from Eq. 26 with $\mathbf{M} = \mathbf{M}_d$
Step 4 Compute $\bar{\boldsymbol{\pi}}^{j+1}$ from Eq. 22c as

$$\bar{\boldsymbol{\pi}}^{j+1} = -\hat{\mathbf{M}}_d^{-1}\mathbf{Q}^{\mathrm{T}}\bar{\mathbf{p}}^{j+1} \tag{29}$$

Step 5 Update the coordinates of the mesh nodes. From the definition of the velocity $v_i = \partial u_i/\partial t$ it is deduced.

$$\mathbf{x}_i^{j+1} = {}^{n}\mathbf{x}_i + \bar{\mathbf{v}}_i^{j+1}\Delta t \tag{30}$$

Step 6 Check the convergence of the velocity and pressure fields. If convergence is achieved move to the next time step, otherwise return to step 1 for the next iteration with $j \leftarrow j + 1$.

Note that solution of steps 1, 3 and 4 does not require the solution of a system of equations as a diagonal form is chosen for $\mathbf{M}$ and $\hat{\mathbf{M}}$.

In the examples presented in the paper the time increment size has been chosen as

$$\Delta t = \min(\Delta t_i) \quad \text{with} \quad \Delta t_i = \frac{h_i^{\min}}{|\mathbf{v}|} \tag{31}$$

where $h_i^{\min}$ is the distance between node i and the closest node in the mesh. Although not explicitely mentioned all matrices and vectors in Eqs. 27–31 are computed at the updated configuration ${}^{n+1}V_F$. This means that the integration domain changes for each iteration and, hence, all the terms involving space derivatives must be updated at each iteration. An alternative is to refer the integrations domain at each time step to the current configuration ${}^{n}V_F$. The jacobian matrix is needed in this case to transform the space derivatives and the differencial of volume from ${}^{n+1}V_F$ to ${}^{n}V_F$ at each iteration.

The boundary conditions are applied as follows. No condition is applied for the computation of the fractional velocities $\bar{\mathbf{v}}^{*}$ in Eq. 25. The prescribed velocities at the boundary are applied when solving for $\bar{\mathbf{v}}^{j+1}$ in step 3.

4 Staggered scheme for the FSI problem

The solution for the variables in the solid and fluid domains at the updated configuration ${}^{n+1}F, {}^{n+1}S$ is found using the staggered scheme shown in Box 1.

LOOP OVER TIME STEPS $n = 1, \cdots n_{\text{time}}$
${}^{n}S \quad {}^{n}F$
LOOP OVER STAGGERED SOLUTION $j = 1, \cdots n_{\text{stag}}$
Solve for solid variables (prescribed tractions at Γ_{FS})
LOOP OVER ITERATIONS $i = 1, \cdots n_{\text{iter}}$
Solve for ${}^{n+1}S_j^i$
Integrate Eq.(3) using a Newmark scheme
Check convergence. Yes: solve for fluid variables
NO: Next iteration $i \leftarrow i + 1$
Solve for fluid variables (prescribed velocities at Γ_{FS})
LOOP OVER ITERATIONS $i = 1, \cdots, n_{\text{iter}}$
Solve for ${}^{n+1}F_j^i$ using the fractional step scheme of Section 3.4
Check convergence. Yes: go to C
Next iteration $i \leftarrow i + 1$
C Check convergence of surface tractions at ${}^{n+1}\Gamma_{FS}$.
Yes: Next time step
Next staggered solution $j \leftarrow j + 1$, $i \leftarrow i + 1$
Next time step ${}^{n+1}S \leftarrow {}^{n+1}S_j^i$, ${}^{n+1}F \leftarrow {}^{n+1}F_j^i$

Box 1 Staggered scheme for the FSI problem

Indeed a ''weak'' version of the staggered scheme can be implemented simply by eliminating the loop over the staggered solution in Box 1. The strong staggered scheme shown in Box 1 is recommended for problems with a large number of solid bodies interacting with the fluid particles. For the bed erosion problems presented in this paper we have used the weak staggered scheme.

5 Treatment of contact between the fluid and a fixed boundary

The motion of the solid is governed by the action of the fluid flow forces induced by the pressure and the viscous stresses acting at the fixed boundary, as mentioned above.

The condition of prescribed velocities at the fixed boundaries in the PFEM are applied in strong form to the boundary nodes. These nodes might belong to fixed external boundaries or to moving boundaries linked to the interacting solids. Contact between the fluid particles and the fixed boundaries is accounted for by the incompressibility condition which naturally prevents the penetration of the fluid nodes into the solid boundaries (Fig. 5). This simple way to treat the fluid–wall contact is another distinct and attractive feature of the PFEM formulation.

6 Generation of a new mesh

One of the key points of the success of the PFEM formulation is the fast regeneration of a mesh at every time step on the basis of the position of the nodes in the space domain. In our work the mesh is *regenerated at each time step* using the so-called extended Delaunay tesselation (EDT) [10, 11, 13]. The EDT generates nonstandard meshes combining elements of arbitrary polyhedrical shapes (triangles, quadrilaterals and other polygons in 2D and tetrahedra, hexahedra and arbitrary polyhedra in 3D) in a computing time of order n, where n is the total number of nodes in the mesh (Fig. 6). The C° continuous shape functions of each element are obtained using the so-called meshless finite element interpolation (MFEM). Details of the mesh generation procedure and the derivation of the MFEM shape functions can be found in [10, 12, 13].

7 Identification of boundary surfaces

One of the main tasks in the PFEM is the correct definition of the boundary domain. Sometimes, boundary nodes are explicitly identified differently

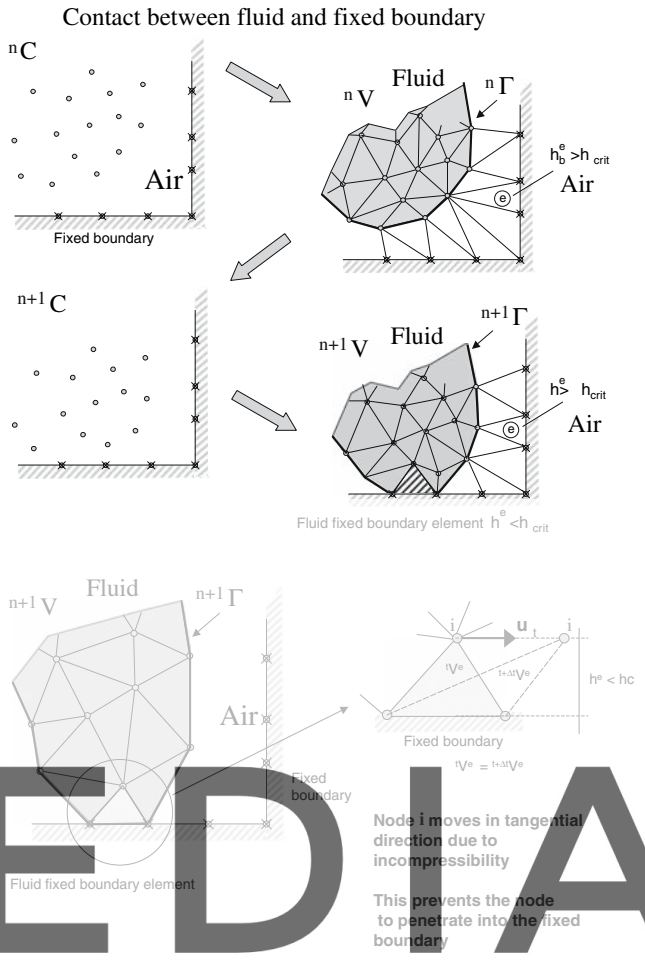

Fig. 5 Automatic treatment of contact condition at the fluid–wall interface

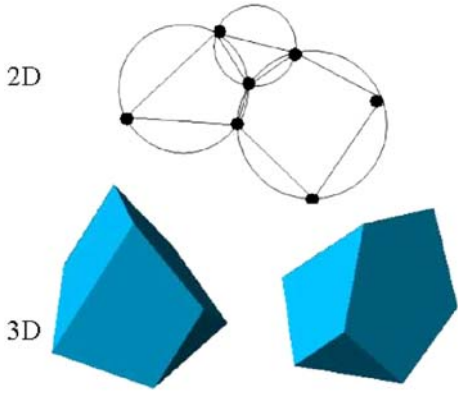

Fig. 6 Generation of nonstandard meshes combining different polygons (in 2D) and polyhedra (in 3D) using the extended Delaunay technique

from internal nodes. In other cases, the total set of nodes is the only information available and the algorithm must recognize the boundary nodes. The extended Delaunay partition makes it easier to recognize boundary nodes. Considering that the nodes follow a variable $h(x)$ distribution, where $h(x)$ is typically the minimum distance between two nodes, the following criterion has been used. All nodes on an empty sphere with a radius greater than αh, are considered as boundary nodes. In practice α is a parameter close to, but greater than one. This criterion is coincident with the Alpha Shape concept [6]. Figure 7 shows an example of the boundary recognition using the Alpha Shape technique. Once a decision has been made concerning which nodes are on the boundaries, the boundary surface and its normal are defined by all the polyhedral surfaces (or polygons in 2D) having all their nodes on the boundary and belonging to just one polyhedron. The boundary definition allows us to compute the volume of each of the fluid and solid subdomains which is also an important task. In the criterion proposed above, the error in the boundary surface definition is proportional to h which is an acceptable error. The method described also allows one to identify isolated fluid particles outside the main fluid domain. These particles are treated as part of the external boundary where the pressure is fixed to the atmospheric value (Fig. 7). We recall that each particle is a material point characterized by the density of the solid or fluid domain to which it belongs. Mass is lost in the analysis domain when a boundary element is eliminated due to departure of a node (a particle) from the domain. This mass is, however, regained when the "flying" node falls down and a new boundary element is created by the Alpha Shape algorithm when the falling node is at a distance less than αh from the boundary. This concept is essential for modeling the splashing of surface waves and bed erosion as described in Sect. 8. An example of wave splashing is presented in Fig. 8 where the motion of a fluid within an oscillating container is shown.

7.1 Contact between solid–solid interfaces

The contact between two solid interfaces can be modeled by introducing a layer of contact elements between the two interacting solid interfaces. This layer is automatically created during the mesh generation by prescribing a minimum distance between two solid boundaries. If the distance exceeds the minimum value, then the generated elements are treated as fluid (or air) elements. Otherwise the elements are treated as contact elements where a relationship between the tangential and normal forces and the corresponding displacement is introduced so as to model elastic and frictional contact in the normal and tangential directions, respectively (Fig. 9).

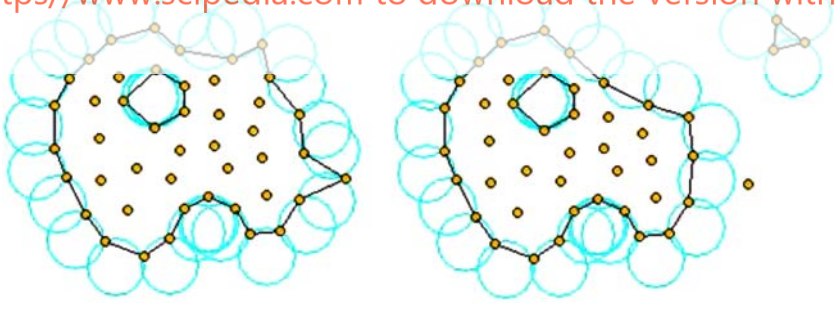

Fig. 7 Identification of individual particles (or a group of particles) starting from a given collection of nodes

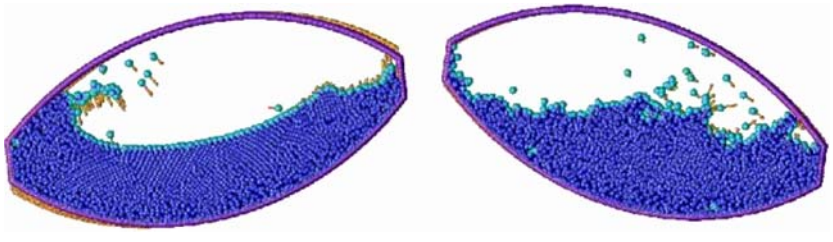

Fig. 8 Motion of a liquid within an oscillating container. Position of the liquid particles at two different times. *Arrows* represent the velocity vector

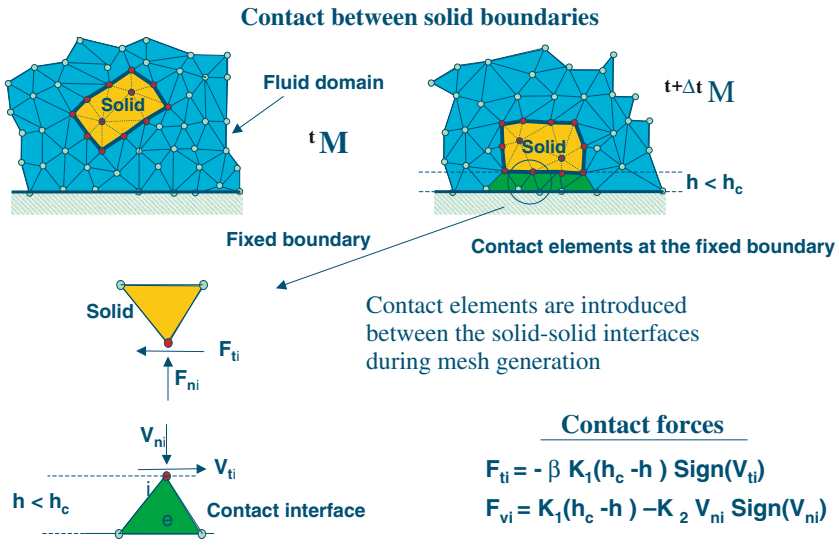

Fig. 9 Contact conditions at a solid–solid interface

This algorithm has proven to be very effective and it allows to identify and model complex frictional contact conditions between two or more interacting solids in an extremely simple manner. The accuracy of this contact model depends on the critical distance above mentioned.

Figure 10 shows an example of the contact algorithm in the analysis of the dragging of a cubic object by a water stream. The contact algorithm described above models accurately the frictional contact effects between the moving cube and the fixed bottom. Other examples of this kind can be found in Oñate et al. [25].

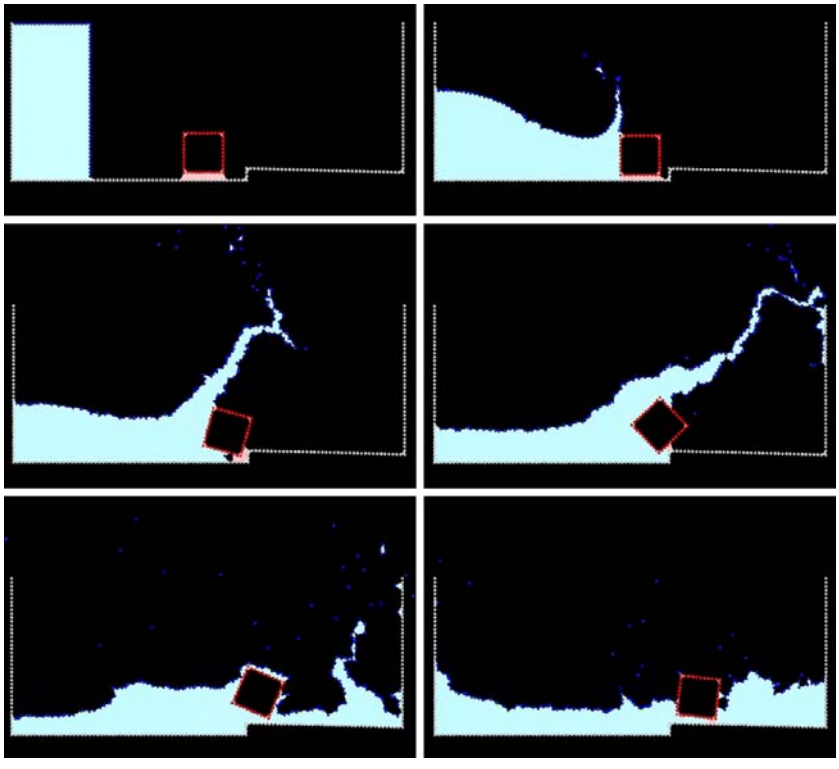

Fig. 10 Dragging of a cubic object by a water stream. Note the contact elements at the cubic bottom interface

8 Modeling of bed erosion

Bed erosion models are traditionally based on a relationship between the rate of erosion and the shear stress level [14, 33]. The effect of water velocity on soil erosion was studied in Parker et al. [26]. In our work we propose a simple erosion model based on the frictional work at the bed surface originated by the shear stresses in the fluid. The resulting erosion model ressembles Archard law typically used for modeling abrasive wear in surfaces under frictional contact conditions [1]. An application of Archard law for modeling surface wear in rock cutting tools can be found in Oñate and Rojek [24]. The algorithm proposed to model the erosion of soil/rock particles at the fluid bed is the following:

1. Compute at every point of the bed surface the resultant tangential stress τ induced by the fluid motion. In 3D problems $\tau = (\tau_{s_n}^2 + \tau_{t_n})^2$ where s and t are the tangential stresses in the plane defined by the normal direction **n** at the bed node. The value of τ for 2D problems can be estimated as follows:

$$\tau_t = \mu \gamma_t \tag{32a}$$

with

$$\gamma_t = \frac{1}{2}\frac{\partial v_t}{\partial n} = \frac{v_t^k}{2h_k} \tag{32b}$$

where v_t^k is the modulus of the tangential velocity at the node k point (i.e. $v_t^k = (v_{s_n}^2 + v_{t_n}^2)^{1/2}$) and h_k is a prescribed distance along the normal of the bed node k. Typically h_k is of the order of magnitude of the smallest fluid element adjacent to node k (Fig. 11).

2. Compute the frictional work originated by the tangential stresses at the bed surface as

$$W_f = \int_0^t \tau_t \gamma_t \, \mathrm{d}t = \int_0^t \frac{\mu}{4}\left(\frac{v_t^k}{h_k}\right)^2 \mathrm{d}t \tag{33}$$

Eq. 33 is integrated in time using a simple scheme as

$$^{n}W_f = {}^{n-1}W_f + \tau \gamma_t \Delta t \tag{34}$$

3. The onset of erosion at a bed point occurs when $^{n}W_f$ exceeds a critical threshold value W_c defined empirically according to the specific properties of the bed material.
4. If $^{n}W_f > W_c$ at a bed node, then the node is detached from the bed region and is allowed to move with the fluid flow, i.e. it becomes a fluid node. As a consequence, the mass of the patch of bed elements surrounding the bed node vanishes in the bed domain and is transferred to the new fluid node. This mass is subsequently transported with the fluid. Conservation of mass of the bed particles within the fluid is guaranteed by changing

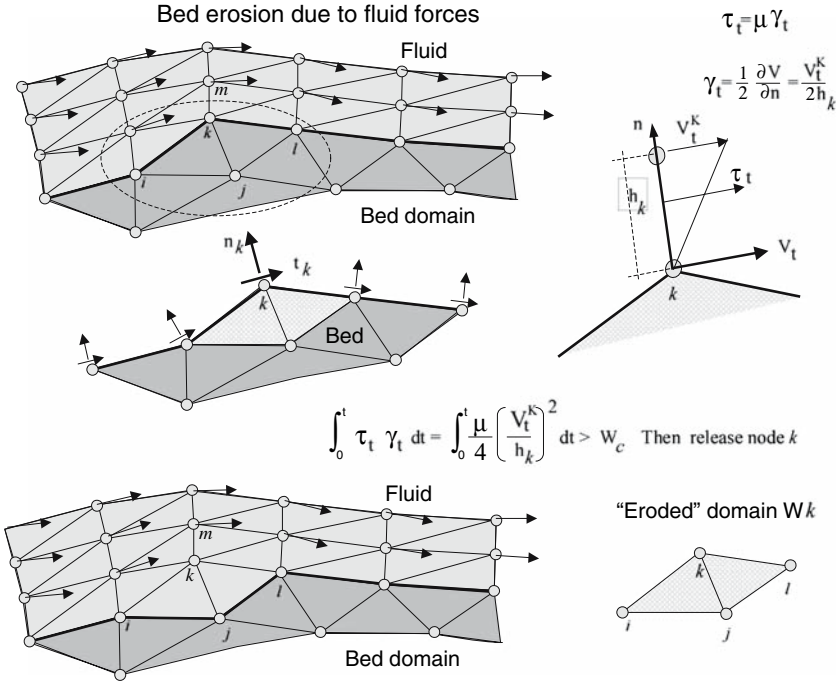

Fig. 11 Modeling of bed erosion by dragging of bed material

the density of the new fluid node so that the mass of the suspended sediment traveling with the fluid equals the mass originally assigned to the bed node. Note that the mass assigned to a node is computed by multiplying the node density by the tributary domain of the node.

5. Sediment deposition is modeled by an inverse process to that described in the previous step. Hence, a suspended node adjacent to the bed surface with a velocity below a threshold value is assigned to the bed surface. This automatically leads to the generation of new bed elements adjacent to the boundary of the bed region. The original mass of the bed region is recovered by adjusting the density of the newly generated bed elements.

Figure 11 shows an schematic view of the bed erosion algorithm proposed.

The examples chosen show the applicability of the PFEM to solve bed erosion problems in free surface flows.

9 Examples

We next present a collection of simple, schematic, but very illustrative two and three dimensional examples showing the potential of the PFEM formulation presented here to model bed erosion in complex free surface flows. Sediment deposition is not considered in any of the examples.

9.1 Example 1. Erosion of a sand hill under a water stream

Figure 12 shows the progressive erosion of a compacted sand domain under the action of an impacting water stream originated by a water jet. The situation is typical in sand shapes built by children in the beach and subsequently destroyed by dropping water on them. The frames in Fig. 12 show the progressive erosion of the surface of the sand domain. A kind of hydraulic jump is generated by the water jet and the sand obstacle as clearly seen in the figures. The erosion process continues until the sand domain is fully dragged by the fluid flow.

9.2 Example 2. Erosion of a 3D earth dam due to an overspill stream

The second example illustrates the erosion of an earth dam under a water stream running over the dam top. A schematic geometry of the dam has been chosen to simplify the computations. The images of Fig. 13 show the progressive erosion of the dam surface until the whole dam is wiped out by the fluid flow.

9.3 Example 3. 3D erosion of a river bed adjacent to a bridge pile

The next example models the progressive erosion of a river bed domain in the vicinity of a bridge pile under a

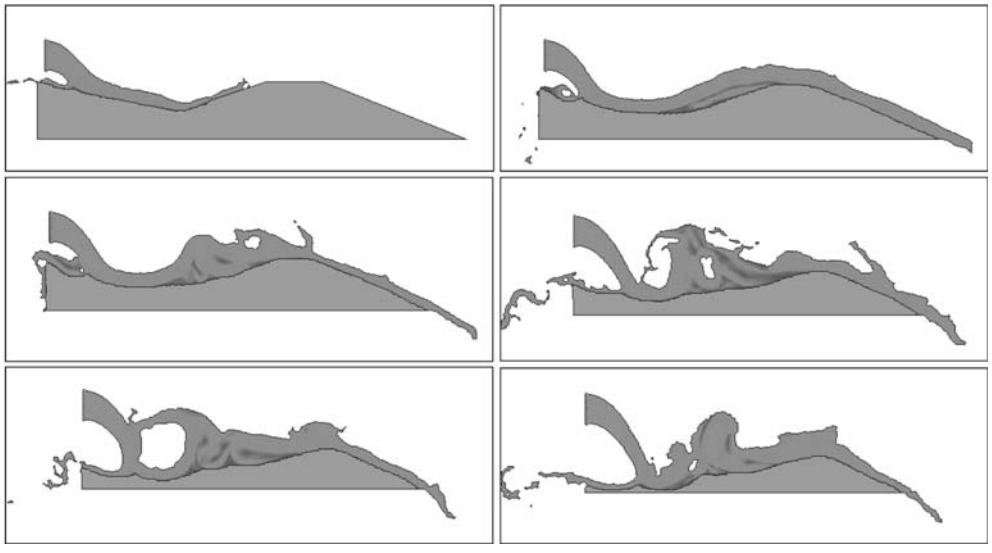

Fig. 12 Erosion of a sand hill due to a water stream

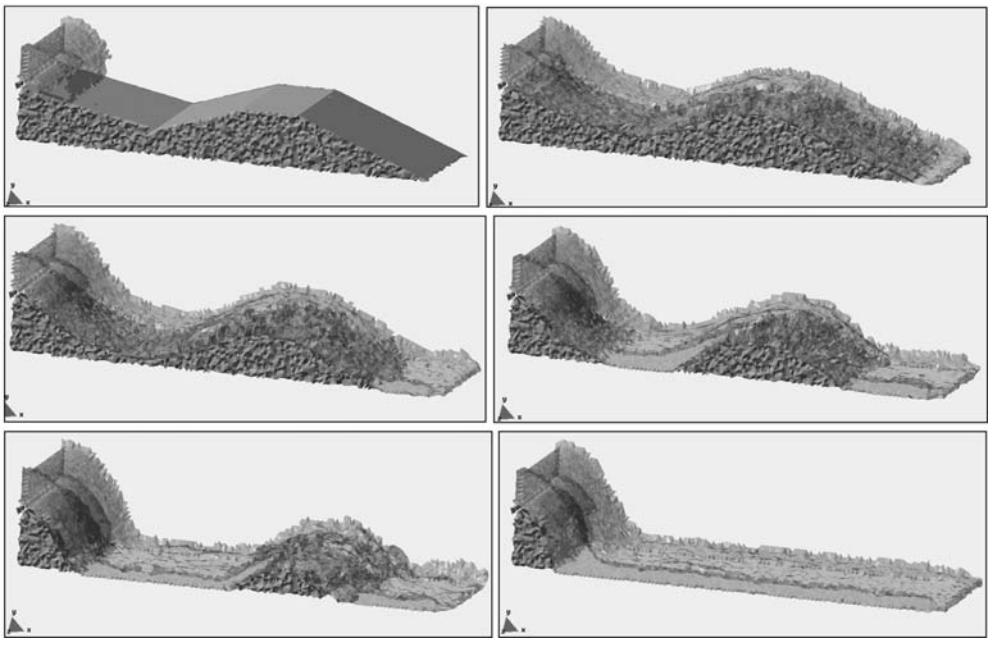

Fig. 13 Erosion of a 3D earth dam due to an overspill stream

water stream. Figure 14 shows a view of the eroded bed surface at different times. The flowing water particles are not shown in the pictures, for clarity. The erosion process continues until the bridge pile foundation is unveiled by the erosion of the adjacent bed particles. We note that the deposition of the eroded particles was not modeled in this case.

9.4 Example 4. Erosion of a solid domain with an object on the top

This final example was chosen so as to demonstrate the effectiveness of the PFEM algorithm to combine the erosion process with the dragging of solid objects. The pictures in Fig. 15 represent schematically a temple on

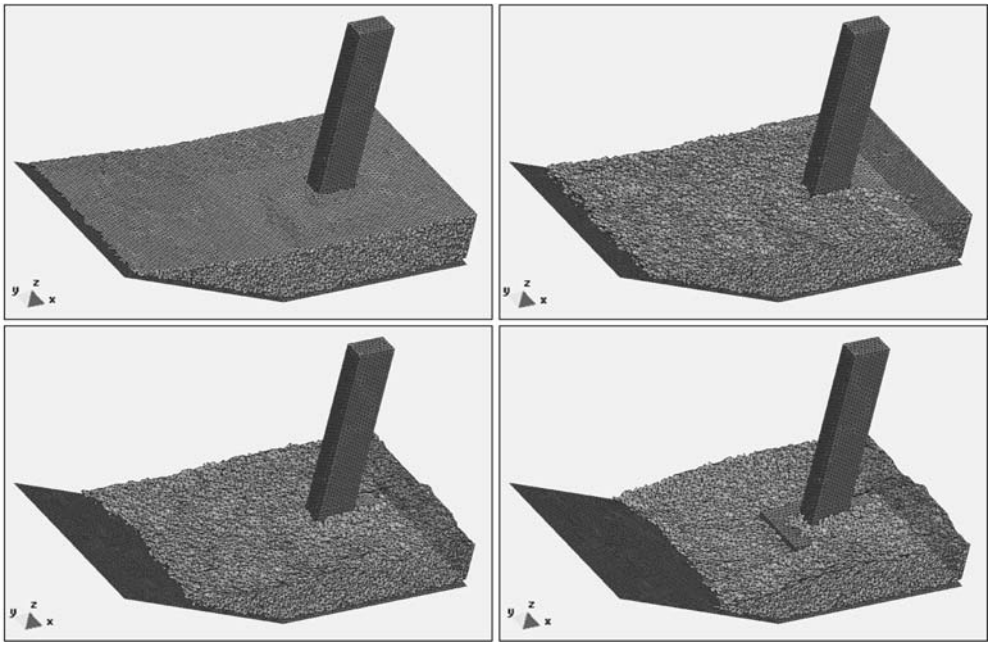

Fig. 14 Evolution of the erosion of the soil in the vicinity of a bridge pile. Water particles are not shown

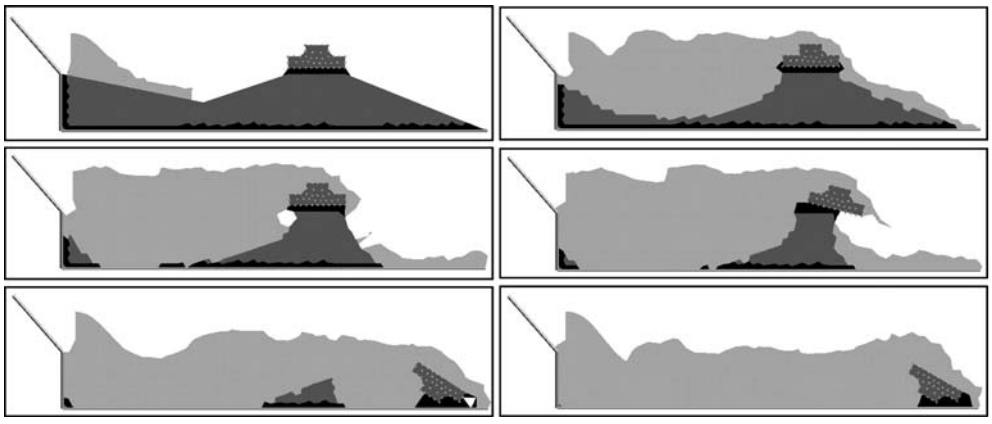

Fig. 15 Erosion of a sand hill with an object on the top

the top of a mountain. The mountain is progressively eroded by a strong water stream until the temple is dragged by the fluid.

10 Conclusions

The PFEM is a powerful technique to model bed erosion problems involving fluids with free surfaces and submerged or floating structures. Problems such as surface erosion, sediment transport and deposition, fluid–structure interaction, large motion of fluid or solid particles, surface waves, water splashing, separation of water drops, etc. can be solved with the PFEM. The success of the method lies in the accurate and efficient solution of the coupled equations for an incompressible fluid and solid dynamics using an updated Lagrangian formulation and a stabilized finite element method. Low order elements with equal order interpolation for all the variables can be effectively used. Other essential solution ingredients are the fast regeneration of the finite element mesh using an extended Delaunay tesselation, the meshless finite element interpolation (MFEM), the identification of the boundary nodes using an Alpha Shape type technique and the simple algorithms to model onset of erosion, sediment transport and material deposition and contact conditions at the fluid–solid and solid–solid interfaces via mesh generation. The examples presented have shown the great potential of the PFEM for modeling bed erosion in complex free surface flows accounting for the dragging of solid objects. Applications of the sediment transport and the material deposition algorithm sketched in this paper will be reported in a forthcoming publication.

Acknowledgments Thanks are given to Dr. R. Rossi and Ms M. de Mier for useful suggestions. This research was partially supported by the FP6 Programme of the European Commission on Global Change and Ecosystems through Project RAMWAS, Project no. FP6-037081. Dr. S.R. Idelsohn is an ICREA Professor at CIMNE.

References

1. Archard JF (1953) Contact and rubbing of flat surfaces. J Appl Phys 24(8):981–988
2. Aubry R, Idelsohn SR, Oñate E (2005) Particle finite element method in fluid mechanics including thermal convection-diffusion. Comput Struct 83(17–18):1459–1475
3. Codina R, Zienkiewicz OC (2002) CBS versus GLS stabilization of the incompressible Navier–Stokes equations and the role of the time step as stabilization parameter. Commun Num Meth Eng 18(2):99–112
4. Darby S, Thorne C (1996) Numerical simulation of widening and bed deformation of straight sand-bed rivers. J Hydr Eng ASCE 122(4):184–193
5. Donea J, Huerta A (2003) Finite element method for flow problems. Wiley, New York
6. Edelsbrunner H, Mucke EP (1999) Three dimensional alpha shapes. ACM Trans Graph 13:43–72
7. Fell R, Wan CF, Cyganiewics J, Foster M (2003) Time for development of internal erosion and piping in embankment dams. J Geotech Geoenviron Eng 129:307–314
8. García J, Oñate E (2003) An unstructured finite element solver for ship hydrodynamic problems. J Appl Mech 70:18–26
9. Idelsohn SR, Oñate E, Del Pin F, Calvo N (2002) Lagrangian formulation: the only way to solve some free-surface fluid mechanics problems. In: Mang HA, Rammerstorfer FG, Eberhardsteiner J (eds) 5th World congress on computational mechanics, July 7–12, Vienna, Austria
10. Idelsohn SR, Oñate E, Calvo N, Del Pin F (2003) The meshless finite element method. Int J Num Meth Eng 58(6):893–912
11. Idelsohn SR, Oñate E, Del Pin F (2003) A Lagrangian meshless finite element method applied to fluid–structure interaction problems. Comput Struct 81:655–671
12. Idelsohn SR, Calvo N, Oñate E (2003) Polyhedrization of an arbitrary point set. Comput Meth Appl Mech Eng 192(22–24):2649–2668
13. Idelsohn SR, Oñate E, Del Pin F (2004) The particle finite element method: a powerful tool to solve incompressible

flows with free-surfaces and breaking waves. Int J Num Meth Eng 61:964–989
14. Kovacs A, Parker G (1994) A new vectorial bedload formulation and its application to the time evolution of straight river channels. J Fluid Mech 267:153–183
15. Oñate E (1998) Derivation of stabilized equations for advective–diffusive transport and fluid flow problems. Comput Meth Appl Mech Engng 151:233–267
16. Oñate E, (2000) A stabilized finite element method for incompressible viscous flows using a finite increment calculus formulation. Comput Meth Appl Mech Eng 182(1–2):355–370
17. Oñate E (2004) Possibilities of finite calculus in computational mechanics. Int J Num Meth Eng 60(1):255–281
18. Oñate E, Idelsohn SR (1998) A mesh free finite point method for advective–diffusive transport and fluid flow problems. Comput Mech 21:283–292
19. Oñate E, García J (2001) A finite element method for fluid–structure interaction with surface waves using a finite calculus formulation. Comput Meth Appl Mech Eng 191:635–660
20. Oñate E, Sacco C, Idelsohn SR (2000) A finite point method for incompressible flow problems. Comput Visual Sci 2:67–75
21. Oñate E, Idelsohn SR, Del Pin F (2003) Lagrangian formulation for incompressible fluids using finite calculus and the finite element method. In: Kuznetsov Y, Neittanmaki P, Pironneau O (eds) Numerical methods for scientific computing variational problems and applications. CIMNE, Barcelona
22. Oñate E, García J, Idelsohn SR (2004) Ship hydrodynamics. In: Stein E, de Borst R, Hughes TJR (eds) Encyclopedia of computational mechanics. Wiley, New York
23. Oñate E, Idelsohn SR, Del Pin F, Aubry R (2004) The particle finite element method. An overview. Int J Comput Methods 1(2):267-307
24. Oñate E, Rojek J (2004) Combination of discrete element and finite element method for dynamic analysis of geomechanic problems. Comput Meth Appl Mech Eng 193:3087–3128
25. Oñate E, Idelsohn SR, Celigueta MA (2006) Lagrangian formulation for fluid–structure interaction problems using the particle finite element method. Bugeda G et al (eds.) CIMNE, Barcelona
26. Parker DB, Michel TG, Smith JL (1995) Compaction and water velocity effects on soil erosion in shallow flow. J Irrigation Drainage Eng 121:170–178
27. Phillips BC, Sutherland AJ (1989) Spatial lag effects in bed load sediment transport. J Hydr Res 24(1):45–56
28. Rahuel JL, Holly FM, Belleudy PJ, Yang G (1989) Modeling of riverbed evolution for bedload sediment mixtures. J Hydr Engrg ASCE 115(1):1521–1542
29. Sekine M, Parker G (1992) Bed-load transport on transverse slope. Int J Hydr Eng ASCE 118(4):513–535
30. Struiksma N, Olesen KW, Flokstra C, Vriend HJ (1985) Bed deformation in curved alluvial channels. J Hydr Res 23:57–79, Delft
31. van Rijn LC (1984) Sediment transport. Part III: bed forms and alluvial roughness. J Hydr Engrg ASCE 110(12): 1733–1754
32. van Rijn LC (1986) Mathematical modeling of suspended sediment in nonuniform flow. J Hydr Eng ASCE 112(6):433–455
33. Wan CF and Fell R (2004) Investigation of erosion of soils in embankment dams. J Geotech Geoenviron Eng 130:373–380
34. Wu W, Rodi W, Wenka T (1997) Three-dimensional calculation of river flow. In: Proceedings of 27th IAHR congress, international association for hydraulic research, Delft
35. Zienkiewicz OC, Taylor RL, Nithiarasu P (2006) The finite element method for fluid dynamics. Elsevier, Amsterdam
36. Zienkiewicz OC, Taylor RL (2005) The finite element method for solid and structural mechanics. Elsevier, Amsterdam